AD-A032 853 STANFORD UNIV CALIF SYSTEMS OPTIMIZATION LAB F/G 12/1
ON A CLASS OF LEAST-ELEMENT COMPLEMENTARITY PROBLEMS.(U)
JUN 76 J PANG F44620-74-C-0079
UNCLASSIFIED SOL-76-10 AFOSR-TR-76-1150 NL

| OF |

AD
A032853

END

DATE
FILMED
1-77

# ON A CLASS OF LEAST-ELEMENT COMPLEMENTARITY PROBLEMS

BY

JONG-SHI PANG

TECHNICAL REPORT SOL 76-10
JUNE 1976

Systems Optimization Laboratory

Department of
Operations
Research

Stanford
University

Stanford
California
94305

# ON A CLASS OF LEAST-ELEMENT COMPLEMENTARITY PROBLEMS

by

Jong-Shi Pang

TECHNICAL REPORT SOL 76-10

June 1976

SYSTEMS OPTIMIZATION LABORATORY
DEPARTMENT OF OPERATIONS RESEARCH
Stanford University
Stanford, California

Research and reproduction of this report were partially supported by the National Science Foundation Grant MCS 71-03341 A04 and the Air Force Office of Scientific Research Contract F 44620 74 C 0079.

# ABSTRACT

The present paper studies ~~the~~ linear complementarity problems of finding vectors $x$ and $y$ in $R_+^n$ such that $c + Dx + y \geq 0$, $b - x \geq 0$ and $x^T(c + Dx + y) = y^T(b - x) = 0$ where $D$ is a Z-matrix and $b \geq 0$. Complementarity problems of this nature which arise, ~~for example,~~ from the minimization of certain quadratic functions subject to upper and lower bounds on the variables. Two least-element characterizations of solutions to the above linear complementarity problem are established first. Next, a new and direct method to solve this class of problems, which depends on the idea of 'least-element solution' is presented. Finally, applications and computational experience with its implementation are discussed.

# 1. INTRODUCTION

The present paper is concerned with the linear complementarity problem of finding vectors $x, y \in R_+^n$ such that

$$(1.1) \qquad \begin{aligned} u &= c + Dx + y \geq 0 \\ v &= b - x \geq 0 \\ u^T x &= v^T y = 0 \end{aligned}$$

where $b \in R_+^n$, $c \in R^n$ and $D \in R^{n \times n}$. We denote problem (1.1) by the triple $(b,c,D)$. If $D$ is symmetric, which incidentally, is not assumed in this paper, then (1.1) is precisely the Kuhn-Tucker optimality conditions for the quadratic program *of finding* a vector $x \in R_+^n$ to

$$(1.2) \qquad \text{minimize} \quad c^T x + \frac{1}{2} x^T D x \quad \text{subject to} \quad x \leq b \ .$$

It is clear that any quadratic program of minimizing a quadratic function subject to upper and lower bounds on the variables can be cast in the form (1.2).

The theory and applications of the linear complementarity problems with Z-matrices (to be defined in the next section), which, with the assumption of symmetry on the matrices, correspond to the minimization of certain quadratic functions subject to lower bounds on the variables, have received much attention in the literature [2], [7], [8], [9], [10], [18], [19], [22]. Recently, many applications of both problems (1.1) and (1.2) have appeared in various contexts; to mention a few, the unilateral Dirichlet problem with two constraints [21] leads to problem (1.1) where $D$ is a Z-matrix; the taut string problem [23] which has its own applications in inventory theory and statistics, is a special case of (1.2) where the matrix $D$ is Minkowski and tridiagonal; and Cheng's salary administration model [3] gives rise to a problem of the form (1.2) where the matrix $D = nI_n - e_n e_n^T$ with $I_n$ the identity matrix of order $n$ and $e_n^T = (1,\ldots,1) \in R^n$. In all these instances (and many others) the matrix $D \in Z$.

Several recent papers ([7], [8], [9], [12], [16], [22]) have demonstrated that many linear complementarity problems have solutions which are "least elements" of subsets of Euclidean space. In an earlier paper [7], R. W. Cottle and the author summarized this least-element aspect of the solutions for various classes of linear complementarity problems. We focused on the class $\mathcal{C}$ of square matrices which was introduced by Mangasarian [14] in formulating the linear complementarity problems as linear programs and which consists of square matrices satisfying the two conditions: (i) $MX = Y$ and (ii) $r^T X + s^T Y > 0$ for some $r, s \in R_+^n$ and where $X$ and $Y$ are Z-matrices. We demonstrated that linear complementarity problems with matrices in $\mathcal{C}$ have solutions which can be obtained as least elements of polyhedral sets and that the class $\mathcal{C}$ contains all the other classes of matrices considered in [14], [15] and in particular, all those previously known classes of matrices ([8], [9], [12], [22]) for which this solution characterization holds. Despite the fact that $C$ contains many interesting matrices, e.g. those enumerated in the last table in [15], not too many realizations of the linear complementarity problem which yield matrices belonging to $C$ have been seen. Of course, as mentioned earlier, complementarity problems with Z-matrices (which clearly belong to $C$) have many applications, e.g. see [18]. Later in the paper, we shall show that the matrix $\begin{pmatrix} D & I \\ -I & 0 \end{pmatrix}$, which appears in the problem (b,c,D), belongs to $C$ if and only if D is a Minkowski matrix.

In many potential applications of problem (1.1) to obtain numerical solutions of partial differential equations in their discretized form, the matrix D is very large, sparse and structured, in addition to being of class Z. See [6]. Special (iterative) methods under various additional assumptions on the matrix D have been proposed and implemented [1], [6], [13], [17], [21 ]. Direct methods like the principal pivoting scheme [4] and Lemke's almost complementarity pivoting scheme [5] do not appear attractive in this instance because, for one thing, they do not take advantage of the special structure that D possesses, thus creating storage difficulties when handling large-scale problems. Cheng [3] and Veinott [23] have described special (direct) methods for solving the salary administration problem and the taut string problem respectively. Unfortunately, their methods are not applicable to the general problem.

Our purposes in this paper are: (i) to establish two least-element aspects of the linear complementarity problem (b,c,D);

(ii) to present a new (direct) method for solving large-scale linear complementarity problems of this class; and (iii) to report our computational experience in solving some realizations of this class of problems using this new method. The plan of the paper is as follows. In the next section, we review some basic terminology, fix our notations and present a result in lattice theory that will be used in later development. In Section 3, we develop the least-element aspects of the problem. In Section 4, we present our proposed method and discuss some of its refinements when it is applied to problems having further structure. In the fifth and final section of this paper, we report our computational experience with the method.

## 2. NOTATIONS, DEFINITIONS AND BASIC RESULTS

We denote by $R_+^n$ the nonnegative orthant of Euclidean n-space. By $R^{n \times m}$, we denote the space of real $n \times m$ matrices. We use $I_n$ to denote the identity matrix of order n and $e_n$ to denote the summation vector $(1,\ldots,1)^T \in R^n$. The real matrix $A \in R^{n \times n}$ is said to be a Z-matrix (P-matrix) if it has nonpositive off-diagonal entries (positive principal minors). We shall call a matrix $A \in R^{n \times n}$ a K-matrix (or a Minkowski matrix) if it is both a Z- and a P-matrix simultaneously. The classes of all real Z-, P- and K-matrices will be denoted by Z, P and K respectively. It is obvious that principal submatrices of Z-, P- and K-matrices are themselves Z-, P- and K-matrices respectively. Proofs of the following characterizations of P- and K-matrices can be found in Fiedler and Pták [11].

Proposition 2.1. (i) A matrix M is a P-matrix if and only if to every vector $x \neq 0$, there exists an index k such that $x_k(Mx)_k > 0$. (ii) Let $M \in Z$. Then $M \in K$ if and only if there exists a vector $x \geq 0$ such that $Mx > 0$.

For a vector $q \in R^n$ and a matrix $M \in R^{n \times n}$, the linear complementarity problem, (q,M) is that of finding $x \in R_+^n$ such that

$$q + Mx \geq 0 \quad \text{and} \quad x^T(q + Mx) = 0 .$$

Note that the problem (b,c,D) is precisely the linear complementarity problem (q,M) with $q = \binom{c}{b}$ and $M = \begin{pmatrix} D & I \\ -I & 0 \end{pmatrix}$. The problem (q,M) is said to be feasible if there exists a vector $x \in R^n_+$ such that $q + Mx \geq 0$. Any such vector is said to be feasible. The set of feasible vectors is called the feasible set. It has been shown (see Samelson et al. [20]) that $(q,M)$ has a unique solution for every $q \in R^n$ if and only if $M \in P$.

Let $M \in R^{n \times n}$ and $I, J \subseteq \{1,\ldots,n\}$. We define

$$M_{IJ} = \begin{bmatrix} m_{i_1 j_1} & \cdots & m_{i_1 j_t} \\ \vdots & & \vdots \\ m_{i_s j_1} & \cdots & m_{i_s j_t} \end{bmatrix}$$

where $I = \{i_1, \ldots, i_s\}$ and $J = \{j_1, \ldots, j_t\}$ with $1 \leq i_1 < \cdots < i_s \leq n$ and $1 \leq j_1 < \cdots < j_t \leq n$. In particular, $M_{II}$ is a principal submatrix of $M$. Similarly, for a vector $q \in R^n$, we define $q_I = (q_{i_1}, \ldots, q_{i_s})^T$.

For the sake of completeness, we briefly review a few concepts in lattice theory and state a theorem which is essential in the least element study of the problem under consideration. The following discussion and a proof of the theorem can be found in Veinott [24] and Pang [16].

A subset $S$ of $R^n$ is called a meet semi-sublattice (of $R^n$) if for every $x, y \in S$, the vector $z \equiv \min(x,y)$ defined by $z_i = \min(x_i,y_i)$ for each $i$, also belongs to $S$. The subset $S$ is bounded below if there exists a vector $x' \in R^n$ such that $x \geq x'$

for every $x \in S$. An element $\bar{x} \in S$ is a least element of $S$ if $x \geq \bar{x}$ for every $x \in S$. It is clear that a least element, if it exists, must be unique.

Theorem 2.2. Let $S$ be a nonempty and closed meet semi-sublattice of $R^n$. Suppose that $S$ is also bounded below. Then $S$ has a least element.

## 3. CONNECTIONS WITH LEAST ELEMENTS

Throughout this paper, we assume $D \in Z$ and $b > 0$. It has been proved that the linear complementarity problem with a Z-matrix has a solution which is the least element of the feasible set, provided that the latter is nonempty. In fact, this assertion follows immediately from Theorem 2.2 by noting that the feasible set is a meet semi-sublattice of $R^n$ satisfying the conditions in the theorem. It is then not difficult to verify that the least element solves the linear complementarity problem. We note that even though $D \in Z$, the matrix $\begin{pmatrix} D & I \\ -I & 0 \end{pmatrix}$ which appears in the problem $(b,c,D)$ does not belong to $Z$. Therefore the above assertion does not apply. Furthermore, the feasible set of the problem, which is defined as the set

$$\{\binom{x}{y} \in R_+^n \times R_+^n : x \leq b \quad \text{and} \quad c + Dx + y \geq 0\}$$

might not itself be a meet semi-sublattice of $R^{2n}$, as easily constructed examples will show. Therefore, Theorem 2.2 cannot be applied to the above set. In this section, we develop two least element aspects of the problem. Both of them will show that the problem possesses a solution which can be obtained from the least element of subsets of Euclidean spaces, one of which is in $R_+^n$ (the x-space) while the other in $R_+^n \times R_+^n$ (the space of the feasible set).

We start our least element study of the problem by observing that, even without the assumption $D \in Z$, a vector $x \in R_+^n$ satisfying $x \leq b$ clearly determines a (unique) vector $y \in R^n$ such that $(x,y)$

solves the problem provided the conditions below are satisfied for each $i = 1,\dots,n$:

(i) $x_i = 0 \Rightarrow (c + Dx)_i \geq 0$

(ii) $b_i > x_i > 0 \Rightarrow (c + Dx)_i = 0$

(iii) $x_i = b_i \Rightarrow (c + Dx)_i \leq 0$ .

With an abuse of language, we also say that a vector $x \in R^n_+$ satisfying $x \leq b$ and conditions (i)-(iii) is a solution with the understanding that the $y$ vector exists such that $(x,y)$ is a real solution.

We now describe a meet semi-sublattice of $R^n$ having a least element which solves the problem. In the next section, we will present an algorithm which actually computes this least element.

Theorem 3.1. Let $D \in Z$ and

$$S = \bigcap_{i=1}^{n} \{x \in R^n_+ : x \leq b;\ x_i < b_i \Rightarrow (c + Dx)_i \geq 0\}$$

Then $S$ is a meet semi-sublattice of $R^n$ and has a least element. Moreover, this least element is a solution for the problem $(b,c,D)$.

Proof: Since intersections of meet semi-sublattices are meet semi-sublattices, it suffices to show that each set

$$S_i = \{x \in R^n_+ : x \leq b;\ x_i < b_i \Rightarrow (c + Dx)_i \geq 0\}$$

is one such for $i = 1,\dots,n$. Let $x, x' \in S_i$ and $x'' = \min(x,x')$. Clearly, $0 \leq x'' \leq b$. Suppose $x''_i < b_i$ and say, $x''_i = x_i$. Then

$$(c + Dx'')_i = c_i + d_{ii}x''_i + \sum_{j \neq i} d_{ij}x''_j$$

$$\geq c_i + d_{ii}x_i + \sum_{j \neq i} d_{ij}x_j \geq 0 \ .$$

Thus $x'' \in S_i$. Therefore, $S$ is indeed a meet semi-sublattice of $R^n$. It is clearly closed and bounded below by 0: it is nonempty because $b \in S$. Hence by Theorem 2.2, it has a least element, say $\bar{x}$. It remains to verify that conditions (ii) and (iii) are satisfied for $\bar{x}$. We omit these proofs because they are similar to the one given in Lemma 3.10 in Cottle and Pang [7]. □

Theorem 3.1 above shows that when $D \in Z$, the problem $(b,c,D)$ has a solution. The next proposition is concerned with the uniqueness of the solution.

Proposition 3.2. Suppose $D \in K$. Then the problem $(b,c,D)$ has a unique solution $(\bar{x},\bar{y})$ where $\bar{x}$ is the least element of $S$ .

Proof: It suffices to establish the uniqueness part. Let $(\bar{\bar{x}},\bar{\bar{y}})$ be another solution. Then for $i = 1,\ldots,n$, we have

$$(\bar{x} - \bar{\bar{x}})_i(\bar{u} - \bar{\bar{u}})_i = (\bar{x} - \bar{\bar{x}})_i[D(\bar{x} - \bar{\bar{x}}) + (\bar{y} - \bar{\bar{y}})]_i$$

$$= (\bar{x} - \bar{\bar{x}})_i \ (D(\bar{x} - \bar{\bar{x}}))_i + (\bar{x} - \bar{\bar{x}})_i(\bar{y} - \bar{\bar{y}})_i \ .$$

Furthermore,

$$(\bar{x} - \bar{\bar{x}})_i(\bar{u} - \bar{\bar{u}})_i = -\bar{u}_i\bar{\bar{x}}_i - \bar{\bar{u}}_i\bar{x}_i \leq 0$$

and

$$\begin{aligned}(\bar{x} - \bar{\bar{x}})_i(\bar{y} - \bar{\bar{y}})_i &= (\bar{\bar{v}} - \bar{v})_i(\bar{y} - \bar{\bar{y}})_i \\ &= \bar{\bar{v}}_i\bar{y}_i + \bar{v}_i\bar{\bar{y}}_i \geq 0 .\end{aligned}$$

Thus $(\bar{x} - \bar{\bar{x}})_i(D(\bar{x} - \bar{\bar{x}}))_i \leq 0$ for each $i$. Hence it follows from Proposition 2.1 that $\bar{x} = \bar{\bar{x}}$. Clearly, $\bar{y} = \bar{\bar{y}}$. This completes the proof. □

Corollary 3.3. Suppose $D \in K$. Let $\bar{x}$ and $\bar{\bar{x}}$ be the (unique) solutions of the problems $(b,c,D)$ and $(c,D)$ respectively. Then $\bar{x} \leq \bar{\bar{x}}$.

Proof: Let $x = \min(\bar{\bar{x}},b)$. Then $x \in S$, for clearly $0 \leq x \leq b$. If $x_i < b_i$, then $x_i = \bar{\bar{x}}_i$, so that

$$\begin{aligned}(c + Dx)_i &= c_i + d_{ii}x_i + \sum_{j \neq i} d_{ij}x_j \\ &\geq c_i + d_{ii}\bar{\bar{x}}_i + \sum_{j \neq i} d_{ij}\bar{\bar{x}}_j \geq 0.\end{aligned}$$

Thus $x \in S$ and since $\bar{x}$ is the least element of $S$, it follows that $\bar{x} \leq x \leq \bar{\bar{x}}$. This establishes the corollary. □

The results above show that the problem $(b,c,D)$ has a solution which can be obtained from the least element of a meet semi-sublattice of $R^n$. In the rest of this section, we study another least-element aspect of the problem. We recall that the feasible set lies in $R_+^{2n}$.

Therefore, the meet semi-sublattice $S$ and the feasible set belong to two different spaces. Here, our objective is to establish that the feasible set itself contains "some" least element, i.e. a least element under a different partial ordering, that solves the problem. To achieve this, we prove the theorem below.

<u>Theorem 3.4</u>. Let $D \in Z$. Then $D \in K$ if and only if $\begin{pmatrix} D & I \\ -I & 0 \end{pmatrix} \in \mathcal{C}$.

<u>Proof</u>: Necessity. If

$$M = \begin{pmatrix} D & I \\ -I & 0 \end{pmatrix}, \qquad X = \begin{pmatrix} I & -I \\ 0 & D \end{pmatrix} \quad \text{and} \quad Y = \begin{pmatrix} D & 0 \\ -I & I \end{pmatrix}$$

then clearly $MX = Y$. It is obvious that $X \in K$ and $Y \in K$. Thus $M \in \mathcal{C}$.

Sufficiency. Suppose $M \equiv \begin{pmatrix} D & I \\ -I & 0 \end{pmatrix} \in \mathcal{C}$. Then there exist Z-matrices $X$, $Y$ and nonnegative vectors $r$, $s$ such that $MX = Y$ and $r^T X + s^T Y > 0$. We may write

$$X = \begin{pmatrix} X_{11} & -X_{12} \\ -X_{21} & X_{22} \end{pmatrix}$$

where $X_{11}$, $X_{22} \in Z$ and $X_{12}$, $X_{21}$ are nonnegative. Then, we have

$$Y = \begin{pmatrix} D & I \\ -I & 0 \end{pmatrix} \begin{pmatrix} X_{11} & -X_{12} \\ -X_{21} & X_{22} \end{pmatrix}$$

$$= \begin{pmatrix} DX_{11} - X_{21} & -DX_{12} + X_{22} \\ -X_{11} & X_{12} \end{pmatrix}$$

Since $Y$ is a Z-matrix, it follows that $X_{11}$ and $X_{12}$ are nonnegative diagonal matrices. We may write $r^T = (r_1^T, r_2^T)$ and $s^T = (s_1^T, s_2^T)$. By an easy calculation, we obtain

$$r^T X + s^T Y = (r_1^T, r_2^T) \begin{pmatrix} X_{11} & -X_{12} \\ -X_{21} & X_{22} \end{pmatrix} + (s_1^T, s_2^T) \begin{pmatrix} DX_{11} - X_{21} & -DX_{12} + X_{22} \\ -X_{11} & X_{12} \end{pmatrix}$$

$$= (r_1^T X_{11} - r_2^T X_{21} + s_1^T DX_{11} - s_1^T X_{21} - s_2^T X_{11}, \quad - r_1^T X_{12} + r_2^T X_{22} - s_1^T DX_{12} + s_1^T X_{22} + s_2^T X_{12}) .$$

Thus, we have

$$(3.1) \qquad r_1^T X_{11} - r_2^T X_{21} + s_1^T DX_{11} - s_1^T X_{21} - s_2^T X_{11} > 0$$

$$(3.2) \qquad -r_1^T X_{12} + r_2^T X_{22} - s_1^T DX_{12} + s_1^T X_{22} + s_2^T X_{12} > 0 .$$

Expression (3.1) implies

$$(r_1^T + s_1^T D - s_2^T) X_{11} > r_2^T X_{21} + s_1^T X_{21} \geq 0 .$$

Thus,

$$r_1^T + s_1^T D - s_2^T > 0 .$$

Expression (3.2) implies

$$(r_2^T + s_1^T) X_{22} > (r_1^T + s_1^T D - s_2^T) X_{12} \geq 0 .$$

Since $X_{22} \in Z$, Proposition 2.1(ii) implies $X_{22} \in K$. The fact that $Y \in Z$ implies $DX_{12} \geq X_{22}$. Since $D \in Z$ and $X_{12}$ is a nonnegative diagonal matrix, it follows that $DX_{12} \in Z$. Proposition 2.1(ii) then implies $DX_{12} \in K$ and thus $D \in K$. This completes the proof. □

The hypothesis $D \in Z$ is necessary in order for the sufficiency part of the Theorem 3.4 to be true. In the following we give an example of a matrix $D$ such that $D \notin Z$ but $\begin{pmatrix} D & I \\ -I & 0 \end{pmatrix} \in C$.

<u>Example:</u> Let $D = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}$. Obviously $D \notin Z$. Let $M = \begin{pmatrix} D & I \\ -I & 0 \end{pmatrix}$. Choose

$$X = \begin{pmatrix} 1 & 0 & -1 & 0 \\ 0 & 1 & 0 & -1 \\ 0 & -1 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \qquad Y = \begin{pmatrix} 1 & 0 & 0 & -1 \\ 0 & 1 & 0 & 0 \\ -1 & 0 & 1 & 0 \\ 0 & -1 & 0 & 1 \end{pmatrix}.$$

Then it is trivial to verify $MX = Y$. Clearly $X, Y \in Z$. If $r^T = (4,2,3,1)$, and $s^T = 0$, then $r^T X + s^T Y > 0$. Therefore $M \in C$.

Using a result (Theorem 3.11) established in [7], we deduce that when $D \in K$, the (unique) solution of the problem $(b,c,D)$ is given by the vector

$$\begin{pmatrix} \bar{x} \\ \bar{y} \end{pmatrix} = \begin{pmatrix} I & -I \\ 0 & D \end{pmatrix} \begin{pmatrix} \bar{u} \\ \bar{v} \end{pmatrix} = \begin{pmatrix} \bar{u} - \bar{v} \\ D\bar{v} \end{pmatrix}$$

where $\begin{pmatrix} \bar{u} \\ \bar{v} \end{pmatrix}$ is the least element of the polyhedral set

$$\{\binom{u}{v} \in R^n \times R^n : c + Du \geq 0,\ b - u + v \geq 0,\ u - v \geq 0,\ Dv \geq 0\} .$$

Furthermore, it was shown in the same reference that the solution $\begin{pmatrix} \bar{x} \\ \bar{y} \end{pmatrix}$ can be obtained by solving the linear program of finding vectors $x, y \in R^n_+$ which

$$\text{(3.3)} \qquad \begin{aligned} \text{minimize} \quad & p^T x + q^T y \\ \text{subject to} \quad & c + Dx + y \geq 0 \\ & b - x \geq 0 \end{aligned}$$

where $p$ and $q$ are vectors satisfying

$$\begin{pmatrix} I & -I \\ 0 & D \end{pmatrix}^T \begin{pmatrix} p \\ q \end{pmatrix} = \begin{pmatrix} r \\ s \end{pmatrix}$$

for some $\binom{r}{s} > 0$. It is then clear that such vectors $p, q$ can be obtained by first setting $p = r$ and then solving $D^T q = p + s$ for $q$.

The discussion above shows that when $D \in K$, it is possible to solve the linear complementarity problem $(b,c,D)$ via the linear program (3.3). The latter can be solved by the Simplex Method (with the upper bounding technique) or the iterative (relaxation) methods for systems of linear inequalities. On the one hand, the Simplex Method being very general, encounters storage difficulties with large scale problems of this nature. On the other hand, our computational experience with one particular iterative method is quite discouraging, even for problems of very small size (see [7]). Although further analysis and clever modification might bring about improvements in the computational performance of these methods, but we believe that another approach is preferable for solving the problem (b,c,D). In the next section, we propose a new and efficient algorithm for solving large-scale linear complementarity problems of this class. It is a direct method which exploits the structure of the problem.

## 4. A FAST ALGORITHM FOR LARGE-SCALE PROBLEMS

Methods for solving the linear complementarity problem (b,c,D) under various assumptions (e.g. symmetry and positive definiteness) on the matrix D have been proposed and implemented. See [6] for a survey. These include both iterative procedures [1], [13], [17], [21] for the general problem and direct methods for some of its particular cases [3], [23]. Typically, in applications of this class of problems to partial differential equations, the matrices D are very large and sparse. The kinds of direct (pivoting) methods that are normally used to solve linear complementarity problems are undesirable because they do not take advantage of the special and sparse structure of the matrices, thus causing storage difficulties when handling large-scale problems. These features (special structure and sparsity) are very important factors motivating the design of a special efficient algorithm. Our purpose, in this section is to present a new (direct) algorithm for large-scale linear complementarity problems of this nature. In the first part of the section, we formulate the algorithm in its general form. In the latter part of the section, we refine the algorithm to solve the subclass of problems having tridiagonal matrices D and a class of variance-minimization problems.

Because of the similarity of our algorithm and Chandrasekaran's algorithm [2] for linear complementarity problems with Z-matrices, we begin by reviewing the statement of the latter algorithm. It can be shown that the solution obtained by this algorithm is the least element of the feasible set [23].

Algorithm I: Chandrasekaran's Algorithm for $(q,M)$ with $M \in Z$.

Step 0. Let $k = 0$ and $x^{(k)} = 0$.

Step 1. Let $I^{(k)} = \{i:(q + Mx^{(k)})_i < 0\}$ and $J^{(k)} = \{1,\ldots,n\}\setminus I^{(k)}$. If $I^{(k)} = \emptyset$ stop. A solution is $x^{(k)}$. Otherwise continue.

Step 2. Let $I = I^{(k)}$ and $J = J^{(k)}$. Solve $M_{II}x_I^{(k+1)} = -q_I$. If this system of equations does not possess a solution, stop. The problem $(q,M)$ is infeasible. Otherwise set $x_J^{(k+1)} = 0$. Replace $k$ by $k+1$ and go to Step 1.

In what follows, we propose a slight modification of the above algorithm for the case $M \in K$.

Algorithm II: Modified Chandrasekaran Algorithm for $(q,M)$ with $M \in K$.

Step 0. Let $k = 0$ and $x^{(0)} = 0$.

Step 1. If $(q + Mx^{(k)}) \geq 0$, stop. The solution is $x^{(k)}$. Otherwise, let $I^{(k)} = \{i:(q + Mx^{(k)})_i \leq 0\}$ and $J^{(k)} = \{1,\ldots,n\}\setminus I^{(k)}$. Continue.

Step 2. Let $I = I^{(k)}$, and $J = J^{(k)}$. Solve $M_{II}x_I^{(k+1)} = - q_I$ and set $x_J^{(k+1)} = 0$. Replace $k$ by $k+1$ and go to Step 1.

The two algorithms differ in the definition of the set $I^{(k)}$. In the former, the set $I^{(k)}$ is defined as $\{i:(q + Mx^{(k)})_i < 0\}$ which, in general is more restrictive than the one defined in the latter.

In order to explain the motivation for our modification, we recall that an essential purpose of Chandrasekaran's algorithm is to identify the set $I^{(+)}$ of all those indices $i$ for which $(q + M\bar{x})_i = 0$ where $\bar{x}$ is a solution (provided that it exists) to the problem. This purpose is achieved by using Chandrasekaran's observation $(q + Mx^{(k)})_i < 0 \Rightarrow \bar{x}_i > 0$ which, together with the complementarity condition, implies $(q + M\bar{x})_i = 0$. Thus the sequence of sets $I^{(k)}$ forms a successive approximation to the desired set $I^{(+)}$. Recognizing this fact, we see that the algorithm may be improved if the sets $I^{(k)}$ can be made to "converge" faster to $I^{(+)}$, or in other words, if fewer systems of equations have to be solved in Step 2. We therefore propose to modify the algorithm slightly by observing that this stronger implication holds, namely, $(q + Mx^{(k)})_i \leq 0 \Rightarrow (q + M\bar{x})_i = 0$, and redefine the set $I^{(k)}$ as in the second algorithm. Note that the implication $(q + Mx^{(k)})_i \leq 0 \Rightarrow \bar{x}_i > 0$ does not necessarily hold. The new definition of $I^{(k)}$ is intended to include as many indices in $I^{(+)}$ as possible. It has the potential advantage of aggregating several systems of equations into a single one, therefore reducing the number of systems to be solved and thus speeding up the termination of the algorithm. We believe that in this way, the overall efficiency of the algorithm will be increased.

The reason we propose the modified version of the algorithm only for the case $M$ belonging to $K$ is in solving the systems of equations $M_{II}x_I^{(k+1)} = -q_I$ in Step 2. With $M \in Z$ and the old definition of the set $I^{(k)}$, it can easily be proved that the submatrix $M_{II}$ is Minkowski, provided that the problem $(q,M)$ is feasible; in this case, the system of equations $M_{II}x_I^{(k+1)} = -q_I$ can be solved very efficiently by both factorization and iterative methods; furthermore, the solution to the system of equations is bound to be nonnegative. On the other hand, with $M$ simply belonging to $Z$ and our new definition of the set $I^{(k)}$, there is no guarantee that the submatrix $M_{II}$ is nonsingular, as an easy example will show, thus the system $M_{II}x_I^{(k+1)} = -q_I$ may have multiple solutions. Although one can then solve the system: $M_{II}x_I^{(k+1)} = -q$ , $x_I^{(k+1)} \geq 0$, but it is not hard to see that so doing will essentially bring one back to the original Algorithm I. Thus nothing much is gained. However if $M \in K$, then each system $M_{II}x_I^{(k+1)} = -q_I$ is guaranteed to have a (unique) nonnegative solution even with the new definition of $I^{(k)}$. Combining the above facts and the advantage (discussed in the last paragraph) of the new definition of $I^{(k)}$, we conclude that the modified algorithm II is best suited for linear complementarity problems with Minkowski matrices. We do not recommend it for problems $(q,M)$ where $M \in Z$ but $M \notin K$.

We are now ready to state and justify our proposed algorithm for solving the problem $(b,c,D)$ with $D \in Z$ and $b > 0$. The algorithm is a finite scheme which requires solving a sequence of nested subproblems of increasing sizes. It starts with an initialization step which determines the first subproblem to be solved. An inner cycle is then reached where the subproblem is actually processed. The solution obtained there is then tested. The procedure either terminates or enters the outer cycle where a new subproblem (of larger size) is determined. The test for solution is repeated and the outer cycle is again reached or else the procedure terminates. Eventually, the algorithm stops in a finite number of steps with a solution to the problem.

**The Algorithm:**

**Step 0** (Initialization). If $c_i \geq 0$ for every $i$, stop. A solution is given by $\bar{x} = 0$. Otherwise, let $t = 0$, $I(t) = \{i : c_i \leq 0\}$ and $J(t) = \{1,\ldots,n\} \setminus I(t)$.

**Step 1** (Inner cycle). Let $I = I(t)$ and solve the linear complementarity problem $P(t)$:

$$\begin{aligned} -(c_I + D_{II}b_I) + D_{II}v_I &\geq 0 \\ v_I &\geq 0 \\ v_I^T[-(c_I + D_{II}b_I) + D_{II}v_I] &= 0 \end{aligned} \tag{4.1}$$

Let $\bar{v}_{I(t)}$ be the least element solution. (See Remark 2 below.)

**Step 2** (Outer cycle). If $c_i + \sum_{j\in I(t)} d_{ij}(b_j - \bar{v}_j) \geq 0$ for every $i \in J(t)$, stop. A solution is given by $\bar{x}_{I(t)} = b_{I(t)} - v_{I(t)}$ and $\bar{x}_{J(t)} = 0$. Otherwise, let

$$I1(t) = \{i \in J(t) : c_i + \sum_{j\in I(t)} d_{ij}(b_j - \bar{v}_j) \leq 0\}.$$

Set $I(t+1) = I(t) \cup I1(t)$, $J(t+1) = \{1,\ldots,n\} \setminus I(t+1)$; replace $t$ by $t+1$ and go to Step 1.

Remark 1. This algorithm produces a solution in solving at most $n$ subproblems of the form (4.1).

Remark 2. Each subproblem $P(t)$ is itself a linear complementarity problem with a Z-matrix, thus the least element solution $\bar{v}_{I(t)}$ exists if $P(t)$ is feasible (to be established soon) and $\bar{v}_{I(t)}$ can be obtained, for example, by Chandrasekaran's algorithm mentioned earlier.

Remark 3. If $c \leq 0$ in the original problem, then the algorithm reduces to solving the single linear complementarity problem $(-(c+Db),D)$ with the Z-matrix $D$, which of course can be solved very efficiently, for example, by Chandrasekaran's algorithm.

Remark 4. The algorithm is similar to Chandrasekaran's algorithm in that they both require solving subproblems of increasing sizes and checking for termination. The algorithms differ in the subproblems. In our algorithm, the subproblems are linear complementarity problems with Z-matrices, whereas in Chandrasekaran's algorithm, the subproblems are systems of linear equations. In a forthcoming report, we will generalize the problem $(b,c,D)$ to allow some or all of the $b_i$'s to be infinity. This generalization obviously includes the present problem and a linear complementarity problem with a Z-matrix as particular cases. We will also propose an efficient algorithm for this generalized problem and show that it unifies the algorithm above and Chandrasekaran's algorithm.

Remark 5. In the forthcoming report, we will demonstrate that the solution generated by the algorithm is precisely the least-element of the meet semi-sublattice $S$ introduced in the last section.

The success of the algorithm relies on the fact that we can eliminate the upper bounding conditions explicitly and produce certain linear complementarity problems with nice matrices which can be processed very efficiently by existing methods. *In order to guarantee that the solutions thus generated satisfy the upper bounding conditions, we require that they be the least-element of the feasible sets. Here, we see how the idea of least-element solution plays an important role in the algorithm.*

In order to establish the validity of the algorithm, it suffices to verify two things, namely, that each subproblem $P(t)$ is feasible and that the least element solution $\bar{v}_{I(t)}$ satisfies the condition: $\bar{v}_{I(t)} \leq b_{I(t)}$ for every cycle $t$. We proceed by induction on $t$. Problem $P(0)$ is obviously feasible: $v_{I(0)} = b_{I(0)}$ is a feasible vector. Thus $\bar{v}_{I(0)}$ is well-defined and it follows from the minimality property that $\bar{v}_{I(0)} \leq b_{I(0)}$. Suppose $P(t)$ is feasible and $\bar{v}_{I(t)} \leq b_{I(t)}$. We show that $(\bar{v}_{I(t)}, b_{I1(t)})$ is feasible for $P(t+1)$. It suffices to verify

$$-\left[\begin{pmatrix} c_{I(t)} \\ c_{I1(t)} \end{pmatrix} + \begin{pmatrix} D_{I(t)I(t)} & D_{I(t)I1(t)} \\ D_{I1(t)I(t)} & D_{I1(t)I1(t)} \end{pmatrix} \begin{pmatrix} b_{I(t)} \\ b_{I1(t)} \end{pmatrix}\right]$$

$$+ \begin{pmatrix} D_{I(t)I(t)} & D_{I(t)I1(t)} \\ D_{I1(t)I(t)} & D_{I1(t)I1(t)} \end{pmatrix} \begin{pmatrix} \bar{v}_{I(t)} \\ b_{I1(t)} \end{pmatrix} \geq 0 .$$

But this is clear from the definitions of $\bar{v}_{I(t)}$ and of index set $I1(t)$. Thus problem $P(t+1)$ is feasible. Finally, $\bar{v}_{I(t+1)} \leq b_{I(t+1)}$, because $\bar{v}_{I(t+1)}$ is the least element of the feasible set of $P(t+1)$. This completes the inductive step and the algorithm is therefore justified.

Monotonicity of the Iterates: The proof above shows that the following inequality is valid:

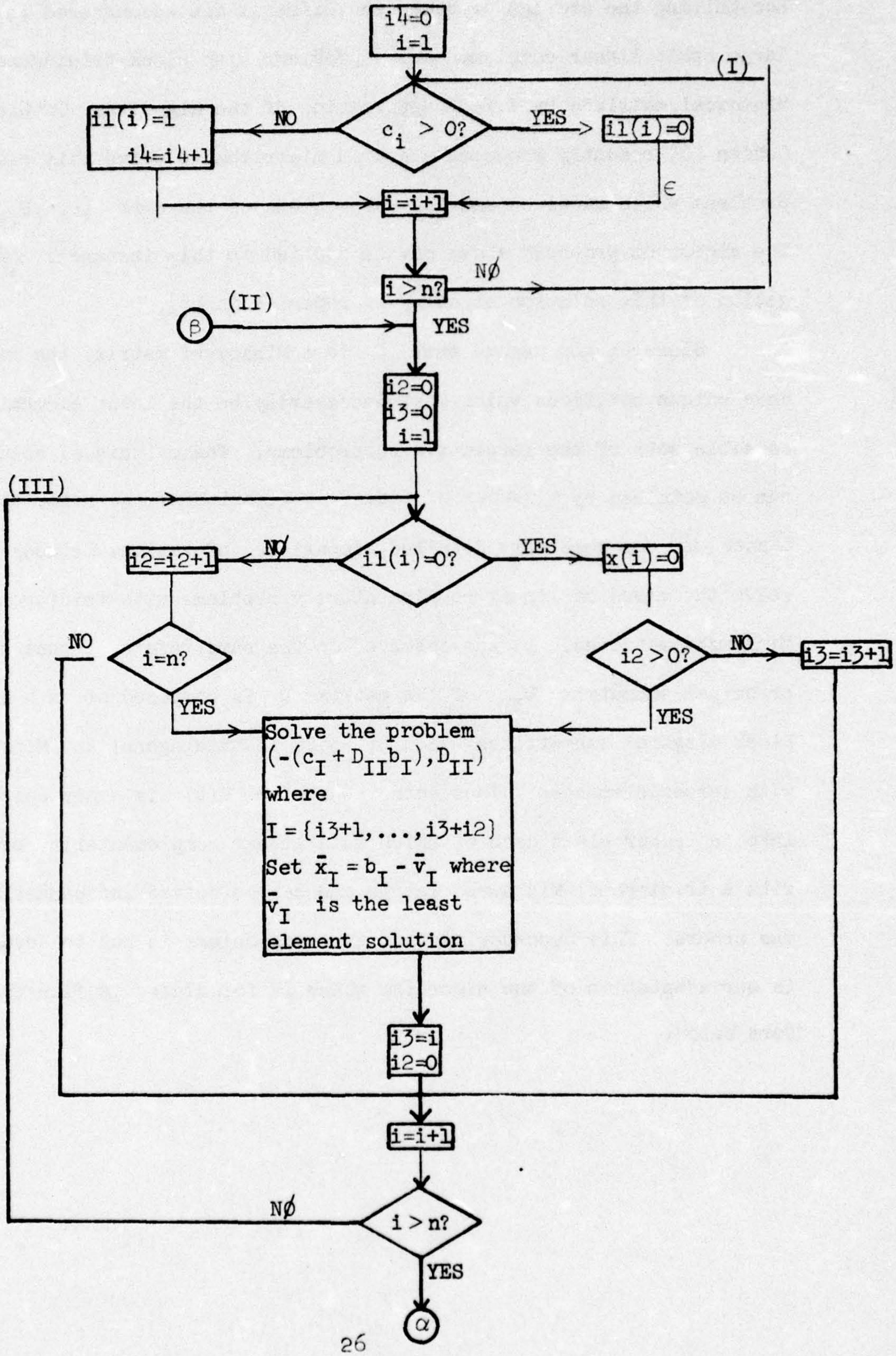
$$\bar{v}_{I(t+1)} \leq \begin{pmatrix} \bar{v}_{I(t)} \\ b_{I1(t)} \end{pmatrix}$$

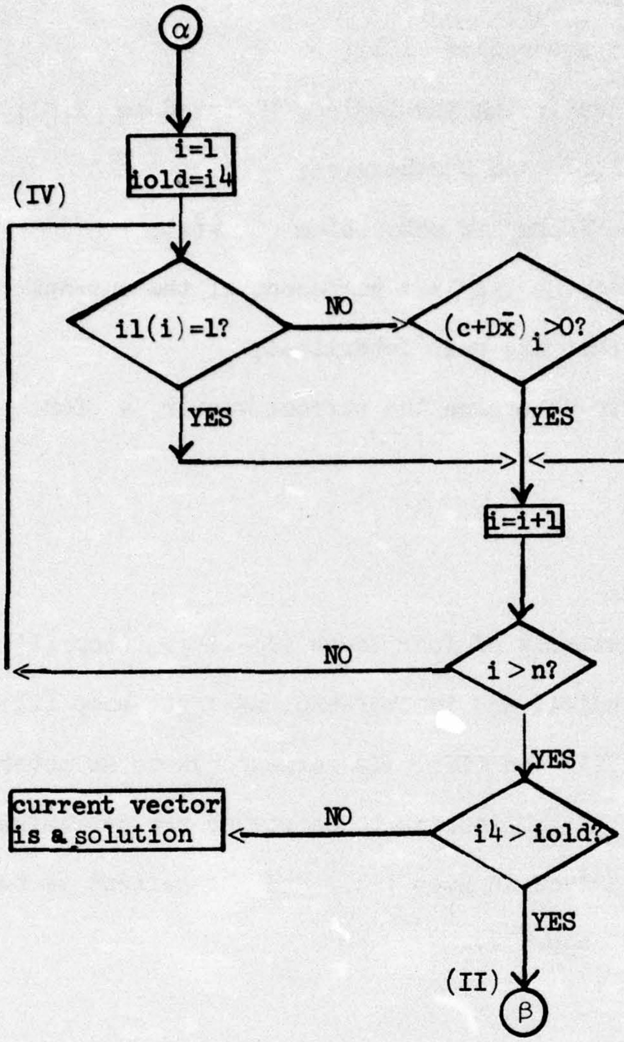
with the understanding that the vector $\bar{v}_{I(t+1)}$ is partitioned in accordance with the vector on the right side. This inequality implies that if some variable $\bar{v}_i$ attains the value 0 at some iteration step t, then it remains at the value 0 in the final solution, thus can be dropped from further consideration. The subsequent subproblems will then have smaller sizes, thus can be solved more quickly.

In order to complete a further analysis of the algorithm, we study some of its refinements when it is applied to solve some particular subclasses of problems. We first consider the case where the matrix D is tridiagonal and Minkowski (i.e. $d_{ij} = 0$ if $|i-j| > 1$). Realizations of this subclass of problems can be found in Veinott's taut string problem [23] and in a discretized version of the unilateral Dirichlet problem with two obstacles [21]. In fact, the matrix D appearing in the latter case has a block tridiagonal structure. This means that the matrix D can be partitioned into blocks $D_{ij}$ $(i, j = 1,\ldots,n)$ where $D_{ij} = 0$ for $|i-j| > 1$. However, the diagonal blocks $D_{ii}$ are tridiagonal.

Recognizing the storage problems and difficulties encountered in solving large scale linear complementarity problems with block tridiagonal Minkowski matrices by direct application of the algorithm, Cottle and Goheen [6] recently proposed a hybrid algorithm to solve this class of problems which requires solving subproblems of the form $(\cdot,\cdot,D_{ii})$. The algorithm proposed above may be applied in this instance. An investigation of this solution strategy is reported in [6].

Since we now assume that $D$ is a Minkowski matrix, the subproblems have unique solutions which must necessarily be the least elements of the feasible sets of the respective subproblems. These (unique) solutions can be obtained by a number of efficient algorithms. We refer to Sacher [18] for some very detailed comparisons of various methods to solve the class of linear complementarity problems with tridiagonal Minkowski matrices. It was observed in the same reference that each principal submatrix $D_{II}$ of the matrix $D$ is composed of $s \geq 1$ block diagonal submatrices (each of which is tridiagonal and Minkowski) with zeroes elsewhere. Thus each subproblem $P(t)$ is again decomposed into $s$ subproblems each of which is a linear complementarity problem with a tridiagonal Minkowski matrix and can be solved independently of the others. This decomposition of the subproblems is put to advantage in our adaptation of the algorithm which is formulated in flow chart form below.


i4=0
i=1
(I)
c$_i$ > 0?
NO
YES
i1(i)=1
i4=i4+1
i1(i)=0
∈
i=i+1
i > n?
NØ
YES
(II)
β
i2=0
i3=0
i=1
(III)
i1(i)=0?
NØ
YES
i2=i2+1
x(i)=0
NO
i=n?
YES
i2 > 0?
NO
YES
i3=i3+1
Solve the problem
$(-(c_I + D_{II}b_I), D_{II})$
where
$I = \{i3+1, \ldots, i3+i2\}$
Set $\bar{x}_I = b_I - \bar{v}_I$ where
$\bar{v}_I$ is the least
element solution
i3=i
i2=0
i=i+1
i > n?
NØ
YES
α


α
i=1
iold=i4
(IV)
il(i)=1?
NO
(c+Dx̄)_i >0?
i4=i4+1
il(i)=1
YES
YES
i=i+1
NO
i > n?
YES
current vector is a solution
NO
i4 > iold?
YES
(II)
β

**Notations in the Flow Chart.**

i4 = size of each subproblem $P(t)$;

i1 = index set identifying the indices included in $I(t)$;

$i1(i) = 1$ means $i \in I(t)$ and 0 otherwise;

i2 = size of each decoupled subproblem of $P(t)$;

i3 = index identifying the last component of the current vector $x$ that has been determined;

iold = index used to determine the current vector $x$ for solution.

**Comments on the Flow Chart.**

It is composed essentially of four loops (I)-(IV). Loop (I) is an initialization of the indices and is self-explanatory. Loop (II) consists of two subloops (III) and (IV). It corresponds to an outer cycle of the method where the subproblems to be solved are determined. Each subproblem $P(t)$ is solved in Loop (III) and the current vector is checked for solution in Loop (IV).

**Storage Considerations.**

The original data b, c, D are stored in (5n-2) double precision numbers, each of which requires 8 bytes. The integer indices are stored as 4-byte numbers. An additional $n$ double precision numbers are required for the solution vector $x$. Therefore the total requirements, excluding those for the subproblems, are approximately $(5n-2) \times 8 + n \times 4 + n \times 8 \cong 52n$ bytes of storage. Depending on the method used to solve the subproblems, extra storage space may range from 0 to 17n bytes. See [18] for more accurate estimates of storage requirements for the subproblems.

We next proceed to another application of the proposed algorithm and solve a quadratic program studied by Cheng [3] in a model of salary administration. The problem is to find a vector $y \in R^n$ to

$$(4.2) \qquad \text{minimize} \quad n \sum_{j=1}^{n} (A_j + B_j y_j)^2 - (\sum_{j=1}^{n} (A_j + B_j y_j))^2$$

$$\text{subject to} \quad \alpha_j \leq y_j \leq \beta_j, \quad j = 1,\ldots,n \; .$$

In the particular application in which the program appeared, the variables $y_j$ represent employee j's compensation, while parameters $A_j$ and $B_j$ are respectively measures of his rewards and performance. All the constants are positive. Making the substitutions, $z_j = A_j + B_j y_j$, $Z_j^0 = A_j + B_j \alpha_j$, and $Z_j^1 = A_j + B_j \beta_j$ we may write problem (4.2) as

$$(4.2)' \qquad \text{minimize} \quad n \sum_{j=1}^{n} z_j^2 - (\sum_{j=1}^{n} z_j)^2$$

$$\text{subject to} \quad Z_j^0 \leq z_j \leq Z_j^1, \quad j = 1,\ldots,n$$

which can be rewritten in vector notations as

$$(4.3) \qquad \text{minimize} \quad z^T Dz \quad \text{subject to} \quad Z^0 \leq z \leq Z^1$$

where $D = nI_n - e_n e_n^T \in Z$ is symmetric, positive semi-definite $z = (z_j)$, $Z^0 = (Z_j^0)$ and $Z^1 = (Z_j^1)$. Therefore the Kuhn-Tucker conditions which are precisely the problem $(b,c,D)$ with $b = Z^1 - Z^0$ and $c = DZ^0$, are necessary and sufficient for global optimality. Note that if $\bar{x}$ solves $(b,c,D)$, then $\bar{z} = \bar{x} + Z^0$ solves (4.3).

In the sequel, we apply the proposed algorithm to solve problem (4.3) by solving the equivalent problem $(b,c,D)$. Although $D$ is singular (thus $D \notin K$), each of its proper principal submatrix is Minkowski, in fact, every such submatrix of order $k < n$ has the form

$$D_k = nI_k - e_k e_k^T$$

and it can be shown by an easy calculation that

$$D_k^{-1} = \frac{1}{n}\left(I_k + \frac{1}{n-k} e_k e_k^T\right) \qquad \text{if } k < n .$$

A typical subproblem in the proposed algorithm for the problem $(b,c,D)$ is to solve the linear complementarity problem

$$\begin{aligned} -(c_I + D_{II} b_I) + D_{II} v_I &\geq 0 \\ v_I &\geq 0 \\ v_I^T[-(c_I + D_{II} b_I) + D_{II} v_I] &= 0 \end{aligned} \tag{4.4}$$

where $I \subseteq \{1,\ldots,n\}$. Due to the "simplicity" of the matrix $D$ and the fact that each of its proper principal submatrix is Minkowski, we choose the modified Chandrasekaran algorithm (Algorithm II) to solve (4.4). By an easy calculation, we deduce that

$$c = DZ^0 = nZ^0 - ( \sum_{i=1}^{n} Z_i^0 ) e_n$$

and

$$\begin{aligned} f_I &\equiv c_I + D_{II} b_I \\ &= nZ_I^0 - ( \sum_{i=1}^{n} Z_i^0 ) e_{|I|} + n(Z_I^1 - Z_I^0) - \sum_{i \in I} (Z_i^1 - Z_i^0) e_{|I|} \\ &= nZ_I^1 - ( \sum_{i \in I} Z_i^1 + \sum_{i \notin I} Z_i^0 ) e_{|I|} , \end{aligned}$$

where $|I|$ denotes the cardinality of the set $I$.

It is then necessary to solve the system of linear equations*

$$(4.5) \qquad D_{I1.I1} v_{I1} = f_{I1}$$

where $I1$ is the subset of $I$ which consists of those indices $i \in I$ for which $f_i \leq 0$.

*Let $I1 \subseteq I$, we denote by $f_{I1}$ the vector $nZ_{I1}^1 - (\sum_{i \in I} Z_i^1 + \sum_{i \notin I} Z_i^0) e_{|I1|}$. Note that $f_{I_1} = c_{I_1} + D_{I_1 I} b_I$ which is not equal to $c_{I_1} + D_{I_1 I_1} b_{I_1}$. This abuse of notation will occur again in later development.

Case 1. $|I1| = n$. Then (4.5) becomes the whole system

$$Dv = f \qquad \text{where} \qquad f = c + Db \; .$$

In this case, a solution vector to the problem (b,c,D) is given by any solution to

$$Dx + c = 0, \qquad 0 \leq x \leq b$$

or

$$(4.6) \qquad D(x + Z^0) = 0, \qquad 0 \leq x \leq Z^1 - Z^0 \; .$$

Since the matrix $D$ has rank n-1 and $De_n = 0$, it follows that such a vector has a representation

$$(4.7) \qquad x = \lambda e_n - Z^0 \qquad \text{where} \qquad Z_i^0 \leq \lambda \leq Z_i^1, \; i = 1,\ldots,n.$$

Therefore, a solution of the problem (4.3) is given $z = \lambda e_n$ and the problem is thus solved.

Case 2. $|I1| < n$. In this case, problem (4.5) has a unique solution:

$$\begin{aligned} v_{I1} &= D^{-1}_{I1,I1} f_{I1} \\ &= \left[ I_{|I1|} + \frac{1}{n-|I1|} e_{|I1|} e^T_{|I1|} \right] \left[ Z^1_{I1} - \frac{1}{n} \Big( \sum_{i \in I} Z^1_i + \sum_{i \notin I} Z^0_i \Big) e_{|I1|} \right] \end{aligned}$$

Thus,

$$(4.8) \qquad v_{I1} = Z^1_{I1} - \frac{\sum\limits_{i\in J1} Z^1_i + \sum\limits_{i\notin I} Z^0_i}{n - |I1|} e_{|I1|} \quad \text{where } J1 = I\setminus I1.$$

By an easy calculation, we obtain

$$(4.9) \qquad -f_{J1} + D_{J1,I1}v_{I1} = -n\left[Z^1_{J1} - \frac{\sum\limits_{i\in J1} Z^1_i + \sum\limits_{i\notin I} Z^0_i}{n - |I1|} e_{|J1|}\right] .$$

We then look for non-positive components in the above vector and augment the index set I1. This latter step is repeated until we arrive at some index set $I1 \subseteq I$ such that either $|I1| = n$, in which case we have obtained a solution to (4.3) and the algorithm stops, or we obtain a solution to problem (4.4) given by $(v_{I1},0)$ where $v_{I1}$ is defined by (4.8). *In the* latter case, a tentative solution vector to the problem (b,c,D) is

$$(4.10) \qquad x_{I1} = -Z^0_{I1} + \frac{\sum\limits_{i\in J1} Z^1_i + \sum\limits_{i\in J} Z^0_i}{n - |I1|} e_{|I1|}$$

$$x_{J1} = b_{J1} \quad \text{and} \quad x_J = 0$$

where

$$J = \{1,\ldots,n\}\setminus I .$$

Again, by an easy calculation, we may deduce

$$c_J + D_{JI}x_I = n\left[Z^0_J - \frac{\sum\limits_{i\in J} Z^0_i + \sum\limits_{i\in J1} Z^1_i}{n - |I1|} e_{|J|}\right] .$$

The current vector $x$ is then checked for solution by identifying any non-positive component in the vector above. The algorithm either stops or continues with an augmented index set $I$.

Summarizing the above analysis, we formulate a procedure for solving problem (4.3).

1. Read in $n$, $Z^0$ and $Z^1$ $(N = \{1,\dots,n\})$.
   Set Iter(# of inner loops) = 1 and Aver 0 $= \frac{1}{n} \sum_{i=1}^{n} Z_i^0$.

2. Determine the set $IO = \{i \in N : Z_i^0 \leq \text{Aver } 0\}$ and $|IO|$.
   Let Aver 1 $= \frac{1}{n} (\sum_{i \in IO} Z_i^1 + \sum_{i \notin IO} Z_j^0)$ and $JO = N \setminus IO$.

3. Determine the set $I1 = \{i \in IO : Z_i^1 \geq \text{Aver } 1\}$ and $|I1|$. Let $J1 = IO \setminus I1$.

4. If $|I1| = n$, find a scalar $\lambda$ satisfying

$$Z_i^0 \leq \lambda \leq Z_i^1 \qquad \text{for} \quad i = 1,\dots,n\ . \tag{4.11}$$

   In this case, a solution is given by $z = \lambda e_n$.

5. If $|I1| < n$, let Aver 2 $= (n - |I1|)^{-1} (\sum_{i \in JO} Z_i^0 + \sum_{i \in J1} Z_i^1)$.
   Determine $I2 = \{j \in J1 : Z_j^1 \geq \text{Aver } 2\}$.

6. If $I2 \neq \emptyset$, replace $I1$ by $I1 \cup I2$ and go to 4.

7. If $I2 = \emptyset$, set $I3 = \{j \in JO : Z_j^0 \leq \text{Aver } 2\}$.

8. If $I3 \neq \emptyset$, replace $I0$ by $I0 \cup I3$. Update Aver 1, and replace Iter by Iter + 1. Then go to 3.

9. If $I3 = \emptyset$, a solution is given by

$$z_{J0} = Z^0_{J0}, \qquad z_{J1} = Z^1_{J1} \quad \text{and} \quad z_{I1} = \text{Aver } 2\ e_{|I1|} .$$

Remark 1. The procedure shows that two cases can occur, namely, (i) a scalar $\lambda$ satisfying condition (4.11) can be found in which case, the vector having all components equal to $\lambda$ is a solution of problem (4.3), and (ii) there exist index sets $S$ (= J0) and $T$ (= J1) such that the vector $z = (z_i)$ where $z_i = Z^0_i$ for $i \in S$, $z_i = Z^1_i$ for $i \in T$ and $z_i = (|S| + |T|)^{-1} (\Sigma_{j\in S} Z^0_j + \Sigma_{j\in T} Z^1_j)$ for $i \in N\setminus(S \cup T)$ solves (4.3).

Remark 2. The algorithm Cheng proposed in [3] for solving the quadratic program (4.3) is essentially a sorting procedure to identify the sets S and T if they exist. A computational comparison of the two algorithms will be reported elsewhere.

## 5. COMPUTATIONAL EXPERIENCE

This section is a report on our computational experience on solving some linear complementarity problems of the nature studied above. Two sets of experiments were performed; the first of which was on problems (b,c,D) with tridiagonal Minkowski matrices* D, whereas the second was on the quadratic program (4.2) studied by Cheng. An essential purpose of these experiments was to test the capability and efficiency of the proposed algorithm in handling large and practical problems. Specifically we wanted to investigate the number of subproblems that needed to be solved, how their sizes grew at each iteration, and the total execution times. In all the experiments described below, the modified Chandrasekaran algorithm (Algorithm II) was used for the subproblems.

When complementarity problems of this kind are solved, there is always the possibility of fixing some variables at their bounds a priori in order to reduce the dimensionality of the problems. This can be achieved via the following implications which can easily be verified:

(i) $c_i + d_{ii}b_i \le 0 \Rightarrow x_i^* = b_i$

(ii) $c_i + \sum\limits_{j \ne i} d_{ij}b_j \ge 0 \Rightarrow x_i^* = 0$

where $x^*$ is an optimal vector. We believe that this pre-processing procedure will increase the efficiency of the algorithm. See [6]. However, in the experiments performed below, this fixing variables at their bounds is not adopted.

*Applications and computational results of the proposed algorithm to solve problems having block tridiagonal Minkowksi matrices are reported in Cottle and Goheen [6].

The flow chart of Section 4 was coded in a FORTRAN subroutine to solve problems (b,c,D) where

$$D = \begin{pmatrix} d & -e & & & & & \\ -e & d & -e & & & & \\ & \cdot & \cdot & \cdot & & & \\ & & \cdot & \cdot & \cdot & & \\ & & & \cdot & \cdot & \cdot & \\ & & & & \cdot & \cdot & -e \\ & & & & & -e & d \end{pmatrix} \in R^{n \times n} \times K .$$

The computation was done using FORTRAN H with Opt = 2 and double precision arithmetic (8-bytes) to avoid round-off errors. The results of the first set of experiments are summarized in Table 1. The vectors b and c were generated according to the rule

$$c_i = -f + g\gamma_i \quad \text{and} \quad b_i = h\delta_i$$

where f, g, h are positive scalars, $\gamma_i$ and $\delta_i$ are randomly generated numbers in (0,1). The following notations are used in the table:

$$I_c^- = \{i : c_i \leq 0\}$$

$$I_\ell^* = \{i : x_i^* = 0\} \quad \text{and} \quad I_u^* = \{i : x_i^* = b_i\}$$

where $x^*$ is the solution generated. The sizes of the subproblems are listed in the fourth column. The size of the first subproblem is always the same as $|I_c^-|$ and therefore is not listed.

| (d,e,n) | (f,g,h) | $\lvert I_c^- \rvert$ | (outer cycles, sizes of subproblems) | $\lvert I_\ell^{(*)} \rvert$ | $\lvert I_u^{(*)} \rvert$ | cpu time (sec) | computer (IBM) |
|---|---|---|---|---|---|---|---|
| $(2,1,10^3)$ | (4,20,5) | 208 | (2,230),(3,231) | 769 | 63 | .14 | 370/168 |
| $(2,1,10^3)$ | (4,20,5) | 193 | (2,218),(3,219) | 781 | 50 | .14 | 370/168 |
| $(2,1,10^3)$ | (4,8,5) | 492 | (2,652),(3,679) | 321 | 275 | .17 | 370/168 |
| $(2,1,10^3)$ | (4,8,5) | 484 | (2,622),(3,632)<br>(4,633) | 367 | 237 | .20 | 370/91 |
| $(2,1,10^3)$ | (14,20,7.5) | 693 | (2,876),(3,890)<br>(4 891) | 109 | 598 | .20 | 370/168 |
| $(2,1,10^3)$ | (14,20,7.5) | 666 | (2,849),(3,861)<br>(4,862) | 138 | 575 | .22 | 370/91 |
| $(4,1,10^3)$ | (4,20,5) | 208 | (2,221) | 779 | 24 | .20 | 370/91 |
| $(4,1,10^3)$ | (4,20,5) | 213 | (2,223) | 777 | 27 | .12 | 370/168 |
| $(4,1,10^3)$ | (15,20,6.5) | 742 | (2,865),(3,868) | 132 | 305 | .19 | 370/168 |
| $(4,1,10^3)$ | (15,20,6.5) | 750 | (2,872),(3,873) | 127 | 330 | .27 | 370/91 |
| $(2,1,\ 2\times 10^3)$ | (9.20,5) | 907 | (2,1139),(3,1161)<br>(4,1162),(5,1163)<br>(6,1164) | 836 | 446 | .47 | 370/168 |

TABLE 1: Problems with tridiagonal Minkowski matrices and randomly generated constant vectors.

The proposed algorithm has a potential weakness in handling problems where it is necessary to solve many subproblems with slowly increasing size. The last experiment indicates that this situation might really occur. Nevertheless, the execution (cpu) times in the table show that the overall performance of the algorithms is very encouraging. Further experiments of the algorithm will be performed and reported elsewhere.

The second set of experiments is concerned with the solution of Cheng's quadratic program (4.2) by the procedure described at the end of Section 4. Several additional features of the procedure are taken into account in its coding. For example, the averages Aver 1 and Aver 2 can be updated fairly easily without computing from scratch. It is observed that

$$\text{Aver } 1_{\text{new}} = \text{Aver } 1_{\text{old}} + \frac{1}{n}\Big(\sum_{i\in I3} Z_i^1 - \sum_{i\in I3} Z_i^0\Big)$$

and

$$\text{sum } 2_{\text{new}} = \text{sum } 2_{\text{old}} - \sum_{i\in I2} Z_i^1$$

where $\text{sum } 2 = \text{Aver } 2 * (n - |I1|)$. The scalar $\lambda$ in step 4 can be chosen to be any number between $\max\limits_{1\le i\le n} Z_i^0$ and $\min\limits_{1\le i\le n} Z_i^1$. Finally, in testing $I2 \neq \emptyset$ and $I3 \neq \emptyset$ in steps 6 and 8 respectively, it suffices to test $|I1|_{\text{new}} > |I1|_{\text{old}}$ and $|I0|_{\text{new}} > |I0|_{\text{old}}$.

The data are generated in the following way: for $j = 1,\ldots,n$, $\alpha_j \in (0,2)$, $A_j \in (0,2)$ and $B_j \in (0,3)$ are randomly chosen in the intervals; $\beta_j = \gamma_j + \alpha_j$ where $\gamma_i \in (0,3)$ is also randomly produced; finally, $Z_j^0 = A_j + B_j\alpha_j$ and $Z_j^1 = A_j + B_j\beta_j$. The outputs of the experiments are summarized in Table 2. Again, the cpu times show very encouraging results. A comparison of the performance of our algorithm and that of Cheng's sorting procedure will be reported elswhere.

| n | iteration steps and sizes of subproblems | | $\lvert I_\ell^*\rvert$ | $\lvert I_u^*\rvert$ | cpu time (sec) |
|---|---|---|---|---|---|
| 1000 | 1st iter | 553 | 315 | 334 | .15 |
| | 2nd iter | 676 | | | |
| | 3rd iter | 685 | | | |
| 1280 | 1st iter | 648 | 501 | 422 | .18 |
| | 2nd iter | 775 | | | |
| | 3rd iter | 779 | | | |
| 2560 | 1st iter | 1558 | 967 | 1482 | .28 |
| | 2nd iter | 1593 | | | |
| 5120 | 1st iter | 2556 | 2467 | 2460 | .57 |
| | 2nd iter | 2652 | | | |
| | 3rd iter | 2653 | | | |
| 10240 | 1st iter | 5900 | 3158 | 3478 | 1.15 |
| | 2nd iter | 6994 | | | |
| | 3rd iter | 7082 | | | |

TABLE 2: Cheng's quadratic program with randomly generated data.

ACKNOWLEDGMENTS. The author wishes to express his deepest gratitude to Professor R. W. Cottle for reading the manuscript and for making many invaluable comments and suggestions; he also wishes to thank Mr. Mark Goheen for suggesting the information presented in the tables.

# BIBLIOGRAPHY

[1] J. Céa and R. Glowinski, "Sur des méthodes d'optimisation par relaxation," Revue Francaise d'Automatique, Informatique et Recherche Opérationnelle R-3 (1973), 5-31.

[2] R. Chandrasekaran, "A special case of the complementary pivot problem," Opsearch 7 (1970), 263-268.

[3] J.C. Cheng, "Analysis of a quantum price model in commodity futures markets and a fair salary administration system," Ph.D. dissertation, Department of Mathematics, M.I.T., Sept. 1975.

[4] R.W. Cottle, "Principal pivoting method for quadratic programming," Mathematics of the Decision Sciences, Part I (G.B. Dantzig and A.F. Veinott, Jr., eds.) Amer. Math. Society, Providence, R.I., 1968.

[5] R.W. Cottle and G.B. Dantzig, "Complementarity pivot theory of mathematical programming," Linear Algebra and Its Applications 1 (1968), 103-125.

[6] R.W. Cottle and M. Goheen, "On a special class of large quadratic programs," Tech. Report SOL 76-7, Systems Optimization Laboratory, Stanford University, April 1976.

[7] R.W. Cottle and J.S. Pang, "On solving linear complementarity problems as linear programs," Tech. Report SOL 76-5, Systems Optimization Laboratory, Stanford University, March 1976.

[8] R.W. Cottle, G.H. Golub and R.S. Sacher, "On the solution of large, structured linear complementarity problems: III," Tech. Report 74-7, Department of Operations Research, Stanford University, 1974.

[9] R.W. Cottle and A.F. Veinott, Jr., "Polyhedral sets having least elements," Math. Programming 3 (1969), 238-249.

[10] C.W. Cryer, "The solution of a quadratic programming problem using systematic overrelaxation," SIAM J. Control 9 (1971), 385-372.

[11] M. Fiedler and V. Pták, "On matrices with nonpositive off-diagonal elements and positive principal minors," Czech. J. Math. 12 (1962), 382-400.

[12] I. Kaneko, "Isotone solutions of the parametric complementarity problems," Tech. Report SOL 75-3, Department of Operations Research, Stanford University, March 1975.

[13] D.P. O'Leary, "Hybrid conjugate gradient algorithms," Ph.D. dissertation, Computer Sciences Department, Stanford University, 1975.

[14] O.L. Mangasarian, "Linear complementarity problems solvable by a single linear program," Tech. Report No. 237, Computer Sciences Department, University of Wisconsin, Madison, Jan. 1975.

[15] O.L. Mangasarian, "Solution of linear complementarity problems by linear programming," in (G.A. Watson, ed.) Lectures in Mathematics, 506, Numerical Analysis, Springer-Verlag, Berlin, Heidelberg-New York,1976, pp. 166-175.

[16] J.S. Pang, "Least element complementarity theory," Ph.D. dissertation, Department of Operations Research, Stanford University, in preparation.

[17] B.T. Polyak, "The conjugate gradient method in extremal problems," U.S.S.R. Computational Mathematics and Mathematical Physics, No. 4 (1969), 94-112.

[18] R.S. Sacher, "On the solution of large, structured complementarity problems," Ph.D. dissertation, Department of Operations Research, Stanford University, 1974.

[19] R. Saigal, "A note on a special linear complementarity problem," Opsearch 7 (1970), 175-183.

[20] H. Samelson, R.M. Thrall and O. Wesler, "A partitioning theorem for Euclidean n-space," Proceedings Amer. Math. Soc. 9 (1958), 805-807.

[21] F. Scarpini, "Some algorithms solving the unilateral Dirichlet problem with two constraints," Calcolo XII, fasc. II, April 1975.

[22] A. Tamir, "The complementarity problem of mathematical programming," Ph.D. dissertation, Department of Operations Research, Case Western Reserve University, June 1973.

[23] A. F. Veinott, Jr., "Least d-majorized network flows with inventory and statistical applications," Management Science 17, No. 9, May 1971.

[24] A.F. Veinott, Jr., unpublished class notes, Department of Operations Research, Stanford University.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

19 REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

~~1. REPORT NUMBER~~

18 AFOSR - TR - 76 - 115Ø

2. GOVT ~~ACCESSION NO.~~

~~3. RECIPIENT'S CATALOG NUMBER~~

9 Technical Rept.

~~4. TITLE (and Subtitle)~~

6 ON A CLASS OF LEAST-ELEMENT COMPLEMENTARITY PROBLEMS.

~~5. TYPE OF REPORT & PERIOD COVERED~~

Interim

6. PERFORMING ORG. REPORT NUMBER

~~7. AUTHOR(s)~~

10 Jong-Shi Pang

14 SOL-76-1Ø

~~8. CONTRACT OR GRANT NUMBER(s)~~

F44620-74-C-ØØ79,

15 NSF-MCS-71-Ø3341

9. PERFORMING ORGANIZATION ~~NAME AND ADDRESS~~

Stanford University
Department of Operations Research
Stanford, CA 94305

~~10. PROGRAM ELEMENT, PROJECT, TASK~~ AREA & WORK UNIT NUMBERS

61102F 9749-06

11. CONTROLLING OFFICE NAME AND ADDRESS

Air Force Office of Scientific Research/NM
Bolling AFB, Washington, DC 20332

~~12. REPORT DATE~~

11 June 1976

~~13. NUMBER OF PAGES~~

43

~~14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)~~

16 9749 17 Ø6 12 48p.

15. SECURITY CLASS. (of this report)

UNCLASSIFIED

15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

least elements
linear complementarity problems
linear programs
quadratic programs
classes of matrices
applications
computational experiences

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

The present paper studies the linear complementarity problem of finding vectors x and y in $R^n_+$ such that $c + Dx + y \geq 0$, $b - x \geq 0$ and $x^T(c + Dx + y) = y^T(b - x) = 0$ where D is a Z-matrix and $b \geq 0$. Complementarity problems of this nature arise, for example, from the minimization

408765

20 Abstract

of certain quadratic functions subject to upper and lower bounds on the variables. Two least-element characterizations of solutions to the above linear complementarity problem are established first. Next, a new and direct method to solve this class of problems, which depends on the idea of "least-element solution" is presented. Finally, applications and computational experience with its implementation are discussed.